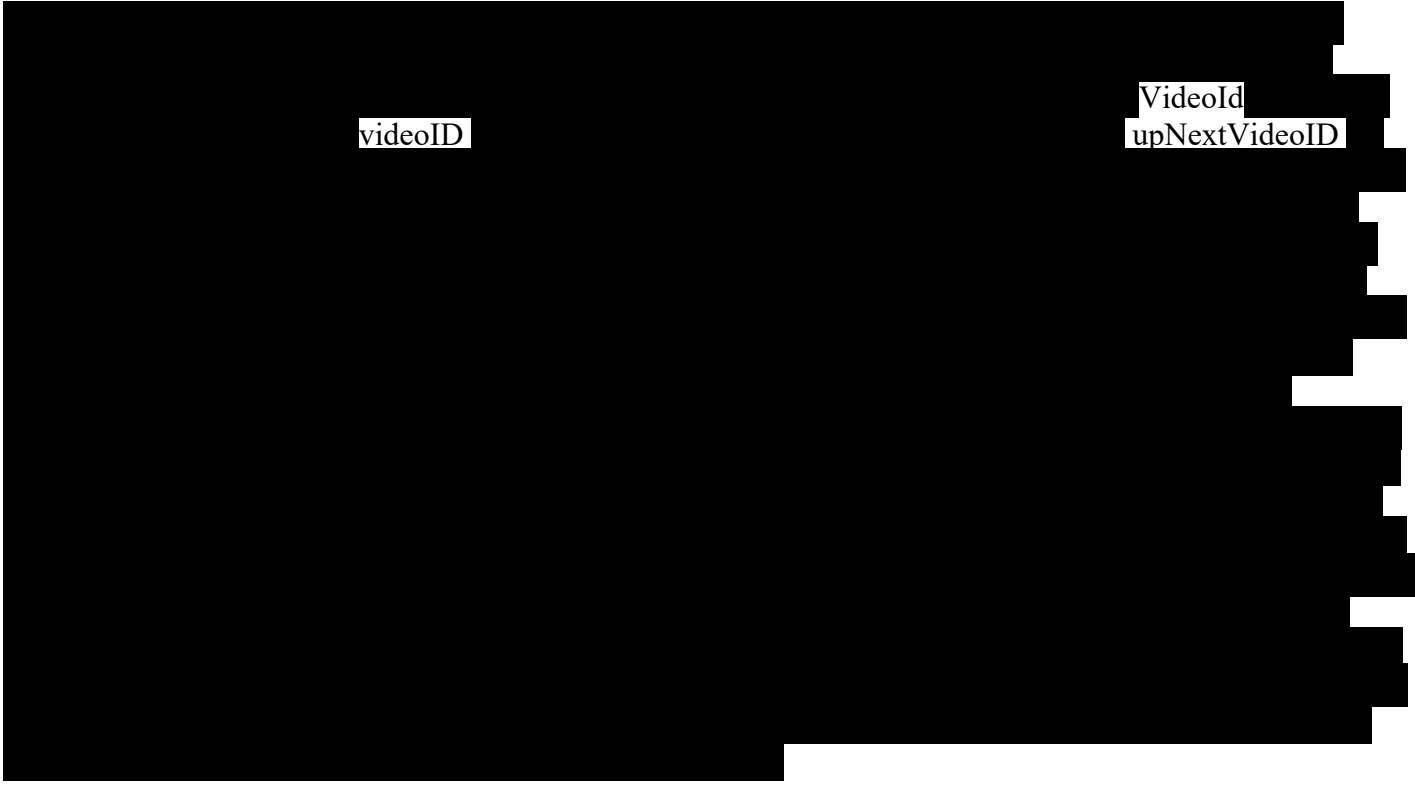# EXHIBIT 3

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	still literally satisfy the "local playback queue on the particular playback device" requirement of limitation 13.6 even if the Court were to adopt Google's proposed construction for that term. videoID VideoId upNextVideoID After proposing the foregoing construction of "local playback queue on the particular playback device" on January 10, 2022, Google shifted course again on February 3, 2022 and for the first time proposed a construction for the term "playback queue" standing alone, which differs from Google's originally proposed construction for "local playback queue on the particular playback device" in several respects. In particular, while Google was previously arguing that "local playback device on the particular playback device" should be construed to mean "a data structure within the particular playback device that maintains an ordered list of two or more multimedia items for playback in the listed order," Google is now taking the position that the

44

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	term "playback queue" standing alone should be construed as "[a]n ordered list of multimedia items that is selected by the user for playback." These two alternate constructions are shown side-by-side below with the language that Google has removed from its prior construction indicated with red highlighting and the language Google has added to its new construction indicated with green highlighting:

1/10/2022 Construction of "local playback queue on the particular playback device"	*2/3/2022 Construction of "playback queue"*
████████████████████████ an ordered list of multimedia items ████████████████ █.	An ordered list of multimedia items that is selected by the user for playback.

Sonos disagrees that this is the proper construction for "playback queue" as that term is used in the context of the '615 Patent and will provide its position regarding the flaws in Google's proposed construction during the claim construction process. However, even if the Court were to adopt such a construction for "playback queue," Sonos maintains that each Cast-enabled computing device installed with any one of the YouTube, YouTube Music, YouTube TV, or YouTube Kids apps would still satisfy claim limitation 13.6(a) literally, or at the very least, under DoE.

As an initial matter, Sonos notes that, unlike its prior construction of "local playback device," Google's latest construction of "playback queue" does not make reference to a "*data structure*," which Google was previously relying on to argue that the "local playback queue" of the '615 Patent is required to be a *singular* "data structure." In this respect, Google appears to have backtracked on its prior attempt to limit the "local playback queue" of the '615 Patent in this way, and now appears to acknowledge that a combination of multiple, individual data variables would fall within the scope of the "local playback device" of the '615 Patent. If Sonos's understanding is correct and Google has dropped its baseless attempt to limit the "local playback device" of the '615 Patent to a *singular* "data structure," then there appears to be no remaining dispute that the "ordered list of multimedia items" aspect of Google's "playback queue" construction is satisfied by each of (i) the "WatchNextResponse" data structure, which comprises an "ordered list of multimedia items" and (ii) the combination of ████████████████████████ ████████████████████ data variables that identify the current media item and the upNextVideoID

45

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	data variable that identifies the next media item, which collectively comprises an "ordered list of multimedia items."
	However, Google has not provided any explanation as to why it changed from the language "a data structure … that maintains an ordered list of two or more multimedia items" to the language "[a]n ordered list of multimedia items," and it is still not clear what impact (if any) Google intends this change to have on the scope of its construction or how Google intends to interpret or apply the phrase "ordered list" in the context of its new construction. To the extent that Google does intend to argue that the "ordered list of multimedia items" aspect of its latest construction is still not satisfied by a combination of multiple, individual variables despite Google's removal of the "data structure" language, Google's only apparent basis for such an argument would be an unduly-narrow interpretation of the phrase "ordered list" that requires a singular data structure, and that argument would fail for very similar reasons to those explained above in connection with Google's singular "data structure" position.
	For instance, as explained above, the "WatchNextResponse" data structure is a singular data structure that is configured to contain an ordered list of multimedia items. Thus, the "WatchNextResponse" data structure by itself (or in combination with one or more of the ██████████████████████████ videoID, and/or upNextVideoID data variables) would still literally satisfy the "ordered list of multimedia items" aspect of Google's latest construction even if Google were to argue for a narrow interpretation of that language.[17]
	Moreover, to the extent Google intends to argue that a combination of multiple, individual data variables that collectively provide an ordered listing of the current and next media items for playback does not literally amount to an "ordered list of multimedia items" based on a theory that this language requires locators of the current and next media items for playback to be maintained together in a singular data structure, the combination of ███████████████████████ videoID data variables for the current media item and (b) the upNextVideoID data variable for the next media item still satisfies Google's proposed construction under DoE. This is because there is merely an insubstantial difference between (i) a

---

[17] As noted previously with respect to Google's proposed construction of "local playback queue," to the extent that Google intends for its latest construction (e.g., by its use of the phrase "multimedia items") to require the "playback queue" to maintain multimedia content in its data form (which Sonos disagrees with), the combination of ████████████████ for the current and next media items alone or in combination with the aforementioned data structure satisfy Google's construction.

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	Cast-enabled media player having a singular data structure that provides an ordered listing of the current and next media items for playback and (ii) a Cast-enabled media player having a combination of multiple, individual data variables that collectively provide an ordered listing of the current and next media items for playback. Indeed, regardless of whether a Cast-enabled media player has a singular data structure that provides an ordered listing of the current and next media items for playback or has separate data variables that collectively provide an ordered listing of the current and next media items for playback in an ordered manner, a Cast-enabled computing device performs the same function (e.g., causing playback of a list of media items to be transferred), in the same way ▮▮▮▮▮▮ ▮▮▮▮ to achieve the same result ▮▮▮▮▮▮. And likewise, regardless of whether a Cast-enabled media player has a singular data structure that provides an ordered listing of the current and next media items for playback or has multiple data variables that collectively provide an ordered listing of the current and next media items for playback in an ordered manner, the Cast-enabled media player is performing substantially the same function (e.g., adding media content to the Cast-enabled media player's "local playback queue"), in substantially the same way ▮▮▮▮▮▮ ▮▮▮▮ to achieve substantially the same result ▮▮▮▮ as a Cast-enabled media player having a singular data structure that provides an ordered listing of the current and next media items for playback. This sort of trivial difference — dividing an "ordered list of multimedia items" for playback across multiple data variables that are intended to collectively identify the current and next media items for playback in sequence — is exactly what DoE is intended to cover.

Given that it is not clear how Google intends to interpret or apply the phrase "ordered list" in the context of its new construction, Sonos also expressly reserves its right to further supplement its infringement contentions if Google later attempts to advance a new interpretation of the phrase "ordered list."

Returning to Google's new construction of "playback queue," Google is also now attempting to add a new requirement that the "ordered list of multimedia items" be ***selected by the user for playback.***" Notably, Google has yet to provide any basis for its position that this new limitation is a required aspect of a "playback queue," and it is still not clear how Google intends to interpret or apply this new limitation in the context of its construction. This is particularly the case given that Google seems to be defining the "local playback |

47

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	queue" – which is a data structure that is configured to contain an identification of whatever media content is queued for playback at a given time ███████████ – in terms of the unrelated details as to how the media items contained within the "local playback queue" were previously selected, which would result in a nonsensical interpretation of the claims where a data structure would qualify as a "local playback queue" during some periods of time (*i.e.*, when it contains user-selected media items) and would not qualify as a "local playback queue" during other periods of time (*i.e.*, when it does not contain user-selected media items) despite the fact that it is the exact same data structure and is being used in the exact same manner to facilitate playback. Nevertheless, Google appears to have imported this new limitation into its construction so that it can later use it as a basis for arguing non-infringement of claim limitations 13.5-13.6. However, even if the Court were to adopt a construction of "playback queue" that includes this new limitation requiring an "ordered list of multimedia items" that is "***selected by the user for playback***," each Cast-enabled computing device installed with any one of the YouTube, YouTube Music, YouTube TV, or YouTube Kids apps would still satisfy claim limitation 13.6(a) literally, or at the very least, under DoE. As established above, when a user inputs a request to transfer playback of media content from a Cast-enabled computing device to a Cast-enabled media player via any one of the YouTube, YouTube Music, YouTube TV, or YouTube Kids apps, the Cast-enabled computing device functions to cause ███████████████ ██████████████████████ "videoID" ███████████████████████ ███████████████████████ to store that "videoID" within the set of data variables that comprises the Cast-enabled media player's "local playback queue" (e.g., within the ████████████████████████████████ data variable as well as the corresponding variable within the "WatchNextResponse" data structure), where that "videoID" ██████████████ ███████████████████. As further established above, as part of the transfer, the Cast-enabled media player also functions to obtain a "videoID" of a next media item for playback and then stores that "videoID" within the set of data variables that comprises the Cast-enabled media player's "local playback queue" as well (e.g., within the upNextVideoId data variable as well as the corresponding variable within the "WatchNextResponse" data structure). Depending on the nature of the media content for which playback is being transferred, the current and next media items that are identified within the Cast-enabled media player's "local playback queue" may take various forms. ███████████████████████████████████ ██

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	And because each Cast-enabled computing device installed with any one of the YouTube, YouTube Music, YouTube TV, or YouTube Kids apps is programmed with the capability to perform the functionality of claim limitation 13.6(a) in at least this first scenario where the Cast-enabled media player's "local playback queue" literally amounts to an "ordered list of multimedia items that is selected by the user for playback," Google cannot credibly dispute that each such Cast-enabled computing device would still literally satisfy claim limitation 13.6(a) for this reason alone. GOOG-SONOSWDTX-00039673

[18] https://support.google.com/youtube/answer/6327615?hl=en [Autoplay videos]; https://support.google.com/youtubekids/answer/6130531?hl=en [Recommended videos]; https://support.google.com/youtubekids/answer/6138623?hl=en&co=GENIE.Platform%3DAndroid [Accessibility on YouTube Kids] ("When autoplay is turned on, we'll automatically play another related video."); GOOG-SONOSWDTX-00005974 ▮▮▮▮ GOOG-SONOSWDTX-00039798 ▮▮▮▮

49

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

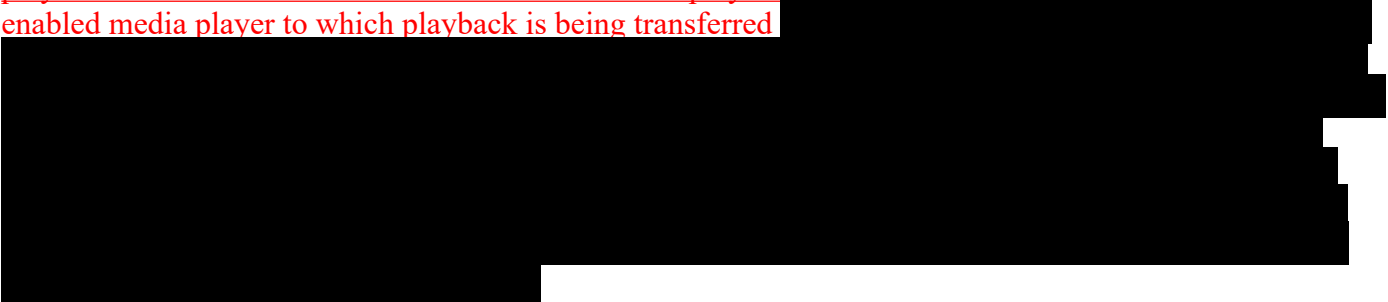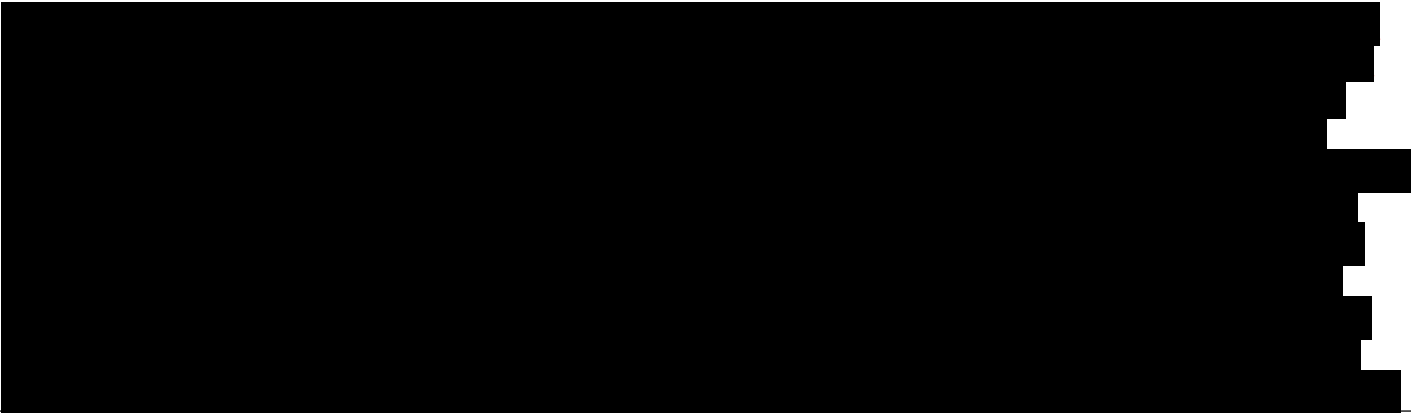| Claim 13	Accused Instrumentalities
	WatchNext response.
	WatchNext response
	see also Infringement Contention Chart for U.S. Patent No. 10,779,033 (Ex. B) at limitation 1.4.
	While it is not entirely clear, it appears that Google may be pursuing this new limitation requiring the "ordered list of multimedia items" to be "selected by the user for playback" so that Google can then argue that this second scenario is excluded from the scope of claim limitation 13.6(a). However, even setting aside the flaws in Google's construction (which will be addressed during the claim construction proceedings), such a non-infringement argument fails for several reasons. *First*, because each Cast-enabled computing device installed with any one of the YouTube, YouTube Music, YouTube TV, or YouTube Kids apps is programmed with the capability to perform the functionality of claim limitation 13.6(a) in at least the first scenario where the Cast-enabled media player's "local playback queue" will exclusively contain user-selected media items and thus literally amounts to an "ordered list of multimedia items that is selected by the user for playback," each such Cast-enabled computing device would still literally satisfy claim limitation 13.6(a) under Google's construction regardless of the fact that there is a second scenario where the Cast-enabled media player's "local playback queue" may contain a media item that is not directly selected by a user. *Second*, even in the second scenario where a Cast-enabled media player's "local playback queue" at the time of transfer

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

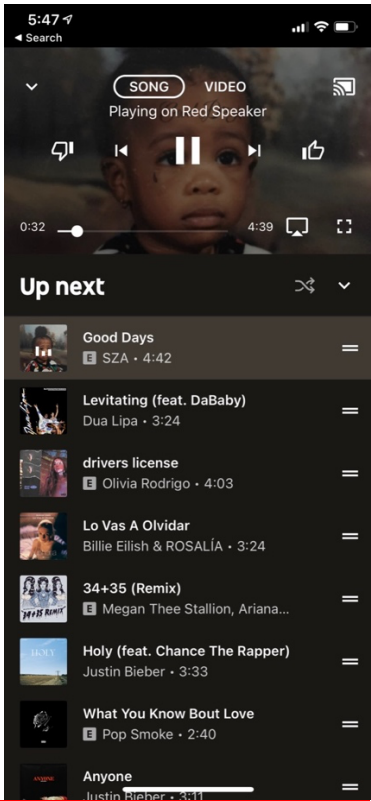| Claim 13	Accused Instrumentalities
	▇▇ the Cast-enabled media player's "local playback queue" still literally amounts to an "ordered list of multimedia items that is selected by the user for playback" because the initial media item in the Cast-enabled media player's "local playback queue" was selected by the user and the additional media item was then identified based on the user's selection. Thus, because each Cast-enabled computing device installed with any one of the YouTube, YouTube Music, YouTube TV, or YouTube Kids apps is programmed with the capability to perform the functionality of claim limitation 13.6(a) in this second scenario where the Cast-enabled media player's "local playback queue" also literally amounts to an "ordered list of multimedia items that is selected by the user for playback," each such Cast-enabled computing device would still literally satisfy claim limitation 13.6(a) for this additional reason. ***Third***, to the extent that the Court adopts Google's construction of "playback queue" and Google then later tries to argue that a Cast-enabled media player's "local playback queue" does not literally amount to an "ordered list that is selected by the user for playback" unless every single media item in the Cast-enabled media player's "local playback queue" is directly selected by the user, the relevant functionality carried out by a Cast-enabled computing device in this second scenario ▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇ ▇▇ ▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇ under DoE. This is because there is merely an insubstantial difference between (i) a Cast-enabled computing device causing playback of media content that was all exclusively selected by a user to be transferred to a Cast-enabled media player such that every media item added to the Cast-enabled media player's "local playback queue" was directly selected by the user and (ii) a Cast-enabled computing device causing playback of media content that was identified based on a user's initial selection of a media item (but was not exclusively selected by the user) to be transferred to a Cast-enabled media player such that only the initial media item added to the Cast-enabled media player's "local playback queue" was directly selected by the user while the other media item was identified based on the user's selection of the initial media item. Indeed, a Cast-enabled computing device performs the same function (e.g., causing playback of media content to be transferred to a Cast-enabled media player), in the same way ▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇ ▇▇▇▇▇▇▇▇▇▇▇▇▇▇ to achieve the same result ▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇ ▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇▇ ***regardless*** of whether the media content for which playback is being transferred was all directly selected by a user or only the initial media item was directly selected by the user and the rest were identified based on the user's selection. And

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

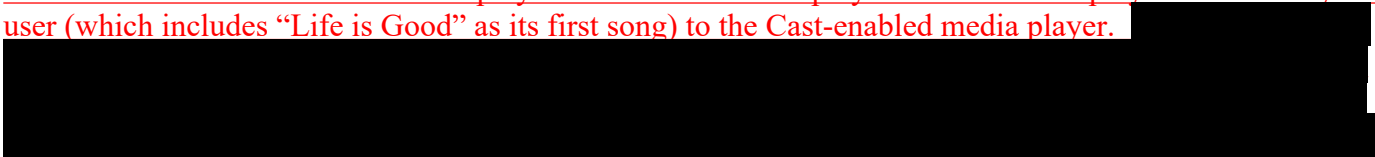| Claim 13	Accused Instrumentalities
	likewise, a Cast-enabled media player performs the same function (e.g., adding media content to the Cast-enabled media player's "local playback queue"), in the same way ███████████████████ ███ ███████████████████ to achieve the same result ███████████ ███ ██████████████████████████████ *regardless* of whether the media content for which playback is being transferred was all directly selected by a user or only the initial media item was directly selected by the user and the rest of the media items were identified based on the user's selection. Lastly, while it is not clear, it appears that Google could also be trying to use its latest construction of "playback queue" to import a temporal limitation into claim limitations 13.5-13.6 requiring that, at the time that a control device detects a set of inputs for transferring playback to a particular playback device, the particular playback device has an "ordered list of multimedia items" that *was already previously populated* with locators for media items that were selected by the *same user that initiated the transfer*. There are two aspects of Google's latest construction of "playback queue" that suggest Google may be planning to make such an argument. First, as noted above, Google has moved away from the language "*a data structure … that maintains* an ordered list of two or more multimedia items" and has switched to the language "[a]n ordered list of multimedia items" without any reference to a "data structure … that maintains" the "ordered list," which suggests that Google may be intending to argue that (i) a "playback queue" cannot exist unless a data structure is *actually populated* with locators of multiple multimedia items and (ii) the media content would therefore have to be added to an "ordered list of multimedia items" that *was already previously populated* with locators for other multimedia items (despite the fact that an "ordered list" can exist in an "empty" state). Second, as noted above, Google has added the requirement that the "ordered list of multimedia items" be "selected by *the* user for playback," and Google's use of the phrase "*the* user" in this clause suggests that Google is trying to import an additional limitation requiring the one or more media items to be added to an "ordered list of multimedia items" that *was already previously populated* with locators for multimedia items that were *selected by the same user that initiated the transfer of playback* (despite the fact that the claim never recites "a user"). Sonos disagrees that this is the proper interpretation of claim limitations 13.5-13.6, both in general and in the context of Google's latest construction for "playback queue," and Sonos intends to address this issue during the claim construction process. However, even if the Court were to interpret claim limitations 13.5-13.6 to

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	have a temporal limitation requiring that, at the time that a control device detects a set of inputs for transferring playback to a particular playback device, the particular playback device has an "ordered list of multimedia items" that ***was already previously populated*** with locators for media items that were selected by the ***same user that initiated the transfer***, Sonos maintains that each Cast-enabled computing device installed with any one of the YouTube, YouTube Music, YouTube TV, or YouTube Kids apps would still satisfy claim limitation 13.6(a) literally, or at the very least, under DoE. As an initial matter, the evidence cited herein establishes that there are various scenarios where the set of data variables comprising the Cast-enabled media player's "local playback queue" will already be populated with locators for media items at the time that a Cast-enabled computing device detects a user's request to transfer playback of media content to the Cast-enabled media player. For example, in one such scenario, the Cast-enabled media player to which playback is being transferred

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	▪▪ The following screenshots illustrate one representative example of this functionality. To begin, the following screenshots illustrate a first Cast-enabled computing device (e.g., a first smartphone or tablet) of a user that has established a first "Cast" session with a particular Cast-enabled media player and thereby transferred playback of a first playlist selected by the user (which includes "Good Days" as its first song) to the Cast-enabled media player:

54

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	At the time that these screenshots were captured, the set of data variables comprising the Cast-enabled media player's "local playback queue" contained locators for at least the current media item (which is "Good Days") and the next media item (which is "Levitating"). While this first "Cast" session is established with the Cast-enabled media player, the same user can then use a second Cast-enabled computing device (e.g., a second smartphone or tablet) to initiate a second "Cast" session with the Cast-enabled media player in order to transfer playback of a second playlist selected by the user (which includes "Life is Good" as its first song) to the Cast-enabled media player. ███████ ███████████████████████████████████████ ███████████████████████████████████████ ███████████████████████████████████████ "local playback queue." This is shown by the following screenshots taken on the second Cast-enabled computing device:

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	  Thus, these screenshots illustrate one possible example where the set of data variables comprising a Cast-enabled media player's "local playback queue" ▮▮▮ Because each Cast-enabled computing device installed with any one of the YouTube, YouTube Music, YouTube TV, or YouTube Kids apps is programmed with the capability to perform the functionality of claim limitation 13.6(a) in scenarios such as this where the set of data variables comprising a Cast-enabled media player's "local playback queue" ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ each such Cast-enabled computing device would still literally satisfy claim limitation 13.6(a) even if Google attempts to rely on its latest construction of "playback queue" to argue that claim limitations 13.5-13.6 include a temporal limitation

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	requiring a playback device to have an "ordered list of multimedia items" that was already previously populated with locators for media items selected by the same user that initiates the claimed transfer of playback to the playback device. Moreover, even in scenarios where the set of data variables comprising the Cast-enabled media player's "local playback queue" are not ██████████████████████████████ ██ the relevant functionality carried out by a Cast-enabled computing device still satisfies claim limitation 13.6(a) under DoE. This is because there is merely an insubstantial difference between (i) a Cast-enabled computing device causing playback of media content to be transferred to a Cast-enabled media player having a "local playback queue" that is already previously populated with locators for media items selected by the user (e.g., as part of a previously-established "Cast" session) and (ii) a Cast-enabled computing device causing playback of media content to be transferred to a Cast-enabled media player having a "local playback queue" that is not already previously populated with locators for media items selected by the user (e.g., a "local playback queue" that is currently empty). Indeed, a Cast-enabled computing device performs the same function (e.g., causing playback of media content to be transferred to a Cast-enabled media player), in the same way ████ ███████████████████████████████████████ ███████████████████████████ *regardless* of whether or not the ███████ ███████████████████████████ And likewise, a Cast-enabled media player performs the same function (e.g., adding media content to the Cast-enabled media player's "local playback queue"), in the same way ████████ ████████████████████████████████████ , to achieve the same result ██████████████ ███████████████ *regardless* of whether or ████████████████████████████ ████████████████████ For all of the foregoing reasons, Sonos maintains that, even if the Court were to adopt Google's latest construction for "playback queue," each Cast-enabled computing device installed with any one of the YouTube, YouTube Music, YouTube TV, or YouTube Kids apps would still satisfy claim limitation 13.6(a) literally, or at the very least, under DoE.

Ex. A –Infringement Contention Chart: U.S. Patent No. 9,967,615
HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

Claim 13	Accused Instrumentalities
	While Sonos has made its best effort to interpret and understand Google's evolving construction of "local playback queue" / "playback queue," and to provide Sonos's infringement position under that evolving construction, it remains unclear how Google intends to interpret and apply that construction to the accused instrumentalities. As such, Sonos expressly reserves the right to further supplement its infringement contentions if Google later attempts to advance an interpretation of this construction that differs from Sonos's current understanding. video id Watch Next video id ○ video id videoId Watch Next Watch Next videoId ;